AMENDMENTS TO THE CLAIMS

1. (Original) A method of automatically generating an updated key value for a segment of keystream for use in a cipher, with forward security, the method comprising the computer-implemented steps of:

receiving a location value that identifies a location of the segment within the keystream;

generating the updated key value corresponding to the identified segment and based on a current key value with forward security and without relying on a key management process for providing such forward secrecy; and

wherein determining another key value based on the current key, the updated key, and state values that are stored during the generating is computationally infeasible.

2. (Original) A method as recited in Claim 1, wherein generating the updated key value further comprises:

creating and storing values in a memory that correspond to a logical tree, wherein the tree represents the keystream, wherein each leaf node of the tree represents a particular keystream segment associated with a discrete location in the keystream, and wherein an order of each leaf node in pre-order traversal of the tree corresponds to a sequential order of all keystream segments.

3. (Original) A method as recited in Claim 2, wherein generating the updated key value further comprises:

creating and storing an ordered plurality of data elements, wherein each of the data elements is identifiable by a node value that is associated with a unique leaf node or intermediate node in the tree, and wherein each of the data elements stores a keystream segment.

4. (Original) A method as recited in Claim 3, wherein generating the updated key value further comprises generating the key value based on the steps of:

selecting a highest-ordered element from among the plurality of data elements;

if the selected element is not associated with a leaf node, then:

storing, in a new highest-ordered element among the plurality of data elements, a first new key value that is determined by applying a first pseudo-random function to the selected element;

generating a second new key value by applying a second pseudo-random function to the selected element.

5. (Original) A method as recited in Claim 4, wherein generating the updated key value further comprises generating the updated key value based on the steps of:

returning, as the updated key value, a segment of the keystream associated with the node identified in the selected next node value when such node is a leaf node;

returning, as the key value that is generated, a segment of the keystream associated with the second new node value when the node identified in the selected next node value is not a leaf node.

6. (Original) A method as recited in Claim 3, wherein generating the updated key value further comprises generating the key value based on the steps of:

determining a current location value;

identifying an internal node of the tree having a highest node number that is an ancestor of a first node corresponding to the received location value and of a second node corresponding to the current location value;

determining a path from the identified internal node to the first node;

traversing the path while applying a first pseudo-random key updating function to the then-current key value during each leftward downward transition and applying a second pseudo-random function during each rightward downward transition.

7. (Original) A method as recited in Claim 6, wherein generating the updated key value further comprises generating the updated key value based on the steps of:

storing, in a new highest-ordered element among the plurality of data elements, each new key value that is generated as part of applying the first and second pseudo-random functions;

generating, as the updated key value, the new key value that stored in the highest-ordered element among the plurality of data elements, when the first node is reached in traversing the path.

8. (Original) A method as recited in Claim 2, wherein each edge of the tree is associated with a distinct pseudo-random function that, when applied to a current key, results in generating a new updated key.

9. (Original) A method as recited in Claim 2, wherein each edge leading leftward and downward from a first node to a second node is associated with a first pseudo-random key updating function, and wherein each edge leading rightward and downward from the first node to a third node is associated with a second pseudo-random key updating function.

10. (Original) A method as recited in Claim 6, wherein each of the pseudo-random functions receives, as input, a first keystream segment and generates, as output, a second keystream segment based on updating the first keystream segment in a pseudo-random manner, such that determining the first keystream segment based on the second keystream segment is computationally infeasible.

11. (Original) A method as recited in Claim 1, further comprising the steps of distributing the updated key value to each member of a multicast group for use in secure communications among the multicast group.

12. (Original) A method of automatically generating an updated key value for a segment of keystream for use in a cipher, with forward security, the method comprising the computer-implemented steps of:
receiving a location value that identifies a location of the segment within the keystream;

generating the updated key value corresponding to the identified segment and based on a current key value with forward security and without relying on a key management process for providing such forward secrecy;

wherein determining another key value based on the current key, the updated key, and state values that are stored during the generating is computationally infeasible;

creating and storing values in a memory that correspond to a logical tree, wherein the tree represents the keystream, wherein each leaf node of the tree represents a particular keystream segment associated with a discrete location in the keystream, and wherein an order of each leaf node in pre-order traversal of the tree corresponds to a sequential order of all keystream segments;

creating and storing an ordered plurality of data elements, wherein each of the data elements is identifiable by a node value that is associated with a unique leaf node or intermediate node in the tree, and wherein each of the data elements stores a keystream segment;

selecting a highest-ordered element from among the plurality of data elements;

if the selected element is not associated with a leaf node, then:

storing, in a new highest-ordered element among the plurality of data elements, a first new key value that is determined by applying a first pseudo-random function to the selected element;

generating a second new key value by applying a second pseudo-random function to the selected element;

returning, as the updated key value, a segment of the keystream associated with the node identified in the selected next node value when such node is a leaf node;

returning, as the key value that is generated, a segment of the keystream associated with the second new node value when the node identified in the selected next node value is not a leaf node;

determining a current location value;

identifying an internal node of the tree having a highest node number that is an ancestor of a first node corresponding to the received location value and of a second node corresponding to the current location value;

determining a path from the identified internal node to the first node;

traversing the path while applying a first pseudo-random key updating function to the then-current key value during each leftward downward transition and applying a second pseudo-random function during each rightward downward transition;

storing, in a new highest-ordered element among the plurality of data elements, each new key value that is generated as part of applying the first and second pseudo-random functions;.

generating, as the updated key value, the new key value that stored in the highest-ordered element among the plurality of data elements, when the first node is reached in traversing the path.

13.-17. (Canceled)

18. (Original) A computer-readable medium carrying one or more sequences of instructions for automatically generating an updated key value for a segment of keystream for use in a cipher, with forward security, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of:

receiving a location value that identifies a location of the segment within the keystream;

generating the updated key value corresponding to the identified segment and based on a current key value with forward security and without relying on a key management process for providing such forward secrecy; and

wherein determining another key value based on the current key, the updated key, and state values that are stored during the generating is computationally infeasible.

19. (Original) An apparatus for automatically generating an updated key value for a segment of keystream for use in a cipher, with forward security, comprising:

means for receiving a location value that identifies a location of the segment within the keystream;

means for generating the updated key value corresponding to the identified segment and based on a current key value with forward security and without relying on a key management process for providing such forward secrecy; and

wherein determining another key value based on the current key, the updated key, and state values that are stored during the generating is computationally infeasible.

20. (Original) An apparatus for automatically generating an updated key value for a segment of keystream for use in a cipher, with forward security, comprising:

a network interface that is coupled to the data network for receiving one or more packet flows therefrom;

a processor;

one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the steps of:

receiving a location value that identifies a location of the segment within the keystream;

generating the updated key value corresponding to the identified segment and based on a current key value with forward security and without relying on a key management process for providing such forward secrecy; and

wherein determining another key value based on the current key, the updated key, and state values that are stored during the generating is computationally infeasible.

21. (New) An apparatus as recited in Claim 19, wherein the means for generating the updated key value further comprises means for creating and storing values in a memory that correspond to a logical tree, wherein the tree represents the keystream, wherein each leaf node of the tree represents a particular keystream segment associated with a discrete location in the keystream, and wherein an order of each leaf node in pre-order traversal of the tree corresponds to a sequential order of all keystream segments.

22. (New) An apparatus as recited in Claim 21, wherein the means for generating the updated key value further comprises means for creating and storing an ordered plurality of data elements, wherein each of the data elements is identifiable by a node value that is associated with a unique leaf node or intermediate node in the tree, and wherein each of the data elements stores a keystream segment.

23. (New) An apparatus as recited in Claim 22, wherein the means for generating the updated key value further comprises means for selecting a highest-ordered element from among the plurality of data elements; means, if the selected element is not associated with a leaf node, for storing, in a new highest-ordered element among the plurality of data elements, a first new key value that is determined by applying a first pseudo-random function to the selected element, and for generating a second new key value by applying a second pseudo-random function to the selected element.

24. (New) An apparatus as recited in Claim 23, wherein the means for generating the updated key value further comprises means for returning, as the updated key value, a segment of the keystream associated with the node identified in the selected next node value when such node is a leaf node; means for returning, as the key value that is generated, a segment of the keystream associated with the second new node value when the node identified in the selected next node value is not a leaf node.

25. (New) An apparatus as recited in Claim 22, wherein generating the updated key value further comprises generating the key value based on the steps of:
determining a current location value;
identifying an internal node of the tree having a highest node number that is an ancestor of a first node corresponding to the received location value and of a second node corresponding to the current location value;
determining a path from the identified internal node to the first node;
traversing the path while applying a first pseudo-random key updating function to the then-current key value during each leftward downward transition and applying a second pseudo-random function during each rightward downward transition.

26. (New) An apparatus as recited in Claim 25, wherein generating the updated key value further comprises means for storing, in a new highest-ordered element among the plurality of data elements, each new key value that is generated as part of applying the first and second pseudo-random functions; means for generating, as the updated key

value, the new key value that stored in the highest-ordered element among the plurality of data elements, when the first node is reached in traversing the path.

27. (New) An apparatus as recited in Claim 21, wherein each edge of the tree is associated with a distinct pseudo-random function that, when applied to a current key, results in generating a new updated key.